Start: March 23,Intermediate: April 30,Done:April 6

Lab 3-4 Report

Hanxi Huang 001303296

Description of Data Structure Algorithms

For the Data structure, I used char array. There are three fields in the class, one is size, one is sign(for detect the positive or negative signs) and the other one is array. I built three constructors. The first one is empty constructor, the second one is used for string, the third one is for an integer.

The constructor for string is check if the array is empty at first. Then, if the sign is positive, create an array contains the string length. After you create the array, just check if the elements are valid(bigger than 0 and smaller than 9, which are all in characters). In the for loop, copy the elements from “val”array to the array. For the negative sign, do the same thing.

The constructor for integer n is using “math.random” method to get a random element.

**Addition:**

Due to I already made a sign to check which sign the element have, I can made the addition in four situations. First step is determine if the two numbers signs are same. If the two numbers have same signs, the sum should be the sum of two numbers. Then, the least significant digit be the sum%10. The carry become sum/10. For example, 999+111. When 9+1, the sum is 10, the carry would be 1. 1 will move to the left(which will add to the next position). The sum will be 0 due to 10%10.

If this array length is larger than h array length, we can do the similar step. When the digits of two numbers are the same, the program do the last for loop. If this length larger, do the inner loop. However, remember to minus 48 due to this is a character array. After finish the inner loop, do plus 48 due to it is a character array. If the h array length is larger than this array length, do the similar thing.

If the two elements have different signs, we can do the following thing: 1.both are positive, the answer would be positive. 2.both are negative, the answer would be negative. 3. This.sign is positive and h.sign is negative, the program invoke this sub h. 4. H.sign is positive and this.sign is negative, the program invoke h sub this.

**Subtract**:

the only case I considered was the case where both were positive.

1. This is positive, h is negative, use add
2. This is negative, h is positive, use add
3. Both negative, use h sub this

**Multiplication:**

I tried to use Karatsuba algorithm to do multiplication. The basic idea is to use recursion. When you multiply two numbers, for example, 12345=12\*1000+345, you need to compute the number by Karatsuba formula. What I did is to create a new multiply method, which use the formula. Then invoke the multiply method in the original multiply method. However, due to my array type is character, it can’t convert to long type in the original method, my try is failed.

**CompareTo:**

Firstly, we need to compute two array length. If one of the array is positive, the other one is negative, it is definitely larger than the other array. If both is negative, we need to measure the length first, if the length are the same, just compare each elements and find which element is larger, the array should be larger.

**Problems**

The biggest problem is that the original code is using the array. Therefore, when doing the calculation, 48 will be added or subtraction. Therefore, it is hard to the calculation. Since if there is one miss calculation, the result will show some not related characters. Furthermore, there are so many the same calculation is needed; therefore, it will cause the biggest problem during the programming and considering the logic.